Oracle Banking Digital Experience

Mobile Application Builder Guide – iOS
Release 19.1.0.0.0

Part No. F18558-01

May 2019

ORACLE®

Mobile Application Builder Guide – iOS

May 2019

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway

Goregaon (East)

Mumbai, Maharashtra 400 063

India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax:+91 22 6718 3001
www.oracle.com/financialservices/

Table of Contents

1. Preface

1.1 Intended Audience

This document is intended for the following audience:

- Customers
- Partners

1.2 Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=accandid=docacc.

1.3 Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

http://www.oracle.com/pls/topic/lookup?ctx=accandid=info or visit

http://www.oracle.com/pls/topic/lookup?ctx=accandid=trs if you are hearing impaired.

1.4 Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual.

The subsequent chapters describes following details:

- Configuration / Installation.

1.5 Related Information Sources

For more information on Oracle Banking Digital Experience Release 19.1.0.0.0, refer to the following documents:

- Oracle Banking Digital Experience Licensing Guide

2. OBDX Servicing Application

2.1 Pre requisite

- Download and Install node js as it is required to run npm and cordova commands.
- XCode to be download from Mac App Store.

2.2 Create Project

1. Extract iOS workspace from installer and place in a folder.
2. The workspace by default contains framework for running on devices. Hence to run the application on simulator, delete and copy the 4 frameworks (OBDXExtensions.framework, OBDXFramework.framework, OBDXWatchFramework.framework and Cordova.framework) from installer/simulator to zigbank\platforms\ios directory.

2.3 Adding UI to workspace

Use any 1 option below

a. Building un built UI (required in case of customizations)

Extract unbuilt UI and traverse to **OBDX_Installer/installables/ui/channel/_build** folder and perform below steps

Windows –

```
npm install -g grunt-cli
npm install
set OBDX_IS_GRUNT=true
node render-requirejs/render-requirejs.js mobile
npm install cwebp-bin
```

Copy "vendor" directory from _build/node_modules/cwebp-bin/ to _build/node_modules/grunt-cwebp/node_modules/cwebp-bin

```
grunt --max_old_space_size=5120 grunt mobilebuild --platform=ios && node
component.js && node integrity-generator.js && node listComponents.js
```

Linux -

```
sudo npm install -g grunt-cli
sudo npm install
export OBDX_IS_GRUNT=true
node render-requirejs/render-requirejs.js mobile
sudo npm install cwebp-bin
```

Copy "vendor" directory from _build/node_modules/cwebp-bin/ to _build/node_modules/grunt-cwebp/node_modules/cwebp-bin

```
node --max_old_space_size=5120 $(which grunt) mobilebuild --platform=ios &&
node component.js && node integrity-generator.js && node listComponents.js
```

b. Using built UI (out of box shipped with installer)

i. Unzip dist.tar.gz for iOS from **OBDX_Mobile\ios\ui** and copy folders(components,extensions,framework,images,json,lzn,home.html ,partials,resource, index.html, build.fingerprint) to workspace (platforms/ios/www/)

Delete originations folder inside images (images/originations) and ensure webhelp folder is not copied.

2.4 Open project in Xcode

Open Xcode by clicking ZigBank.xcodeproj at zigbank/platforms/ios/

1. Adding URLs to app.plist (ZigBank/Resources)

    a. FOR NONOAM (DB Authenticator setup)

SERVER_TYPE	NONOAM
KEY_SERVER_URL	https://mumaa012.in.oracle.com:18443/
WEB_URL	https://mumaa012.in.oracle.com:18443/

    b. OAM Setup (Refer to installer pre requisite documents for OAuth configurations)

SERVER_TYPE	OAM
KEY_SERVER_URL	Eg. https://mumaa012.in.oracle.com:18443/ (This URL must be of OHS without webgate)
WEB_URL	Eg. https://mumaa012.in.oracle.com:18443/
KEY_OAUTH_PROVIDER_URL	http://mum00aon.in.oracle.com:14100/oauth2/rest/token
APP_CLIENT_ID	<Base64 of clientid:secret> of Mobile App client
APP_DOMAIN	OBDXMobileAppDomain
WATCH_CLIENT_ID	<Base64 of clientid:secret> of wearables
WATCH_DOMAIN	OBDXWearDomain
SNAPSHOT_CLIENT_ID	<Base64 of clientid:secret> of snapshot
SNAPSHOT_DOMAIN	OBDXSnapshotDomain
LOGIN_SCOPE	OBDXMobileAppResServer.OBDXLoginScope

    c. IDCS Setup

SERVER_TYPE	IDCS
KEY_SERVER_URL	Eg. https://mumaa012.in.oracle.com:18443/ (This URL must be of OHS without webgate)
WEB_URL	Eg. https://mumaa012.in.oracle.com:18443/
KEY_OAUTH_PROVIDER_URL	http://obdx-tenant01.identity.c9dev0.oc9qadev.com/oauth2/v1/token

APP_CLIENT_ID	<Base64 of clientid:secret> of Mobile App client
WATCH_CLIENT_ID	<Base64 of clientid:secret> of wearables
SNAPSHOT_CLIENT_ID	<Base64 of clientid:secret> of snapshot
LOGIN_SCOPE	obdxLoginScope
OFFLINE_SCOPE	urn:opc:idm:__myscopes__ offline_access

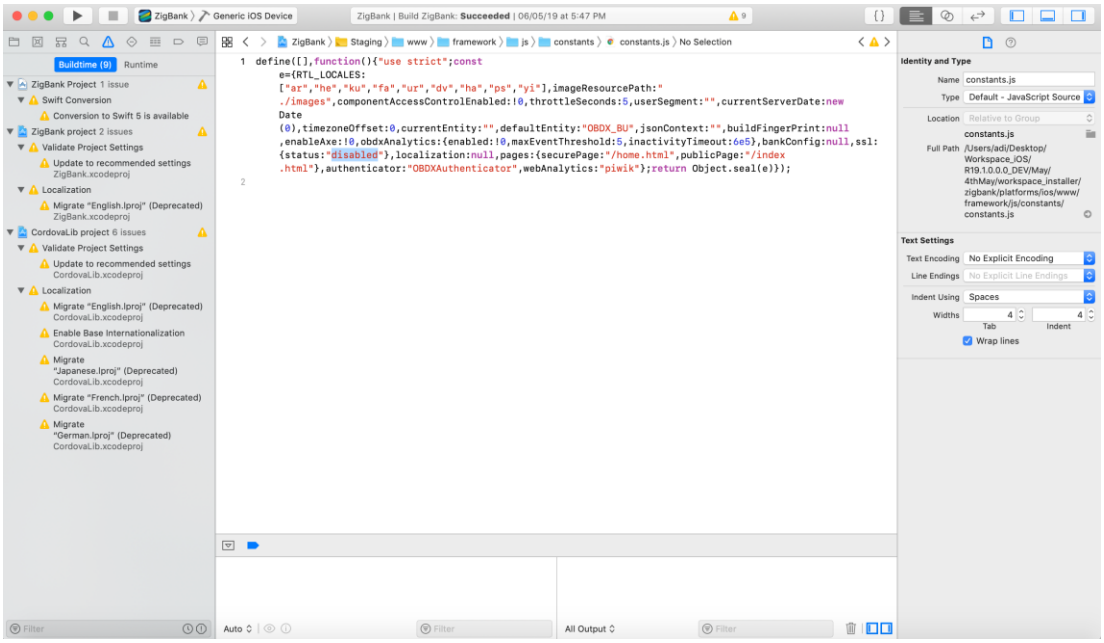
d. Common Configurations

CurrencyCode	Currency code for Siri Payments
PaymentPurposeRequiredFlag	Payment purpose required for Siri payments
SUITENAME	Group identifier for sharing keystore information. Same as given in app groups (mandatory to be given same as App Group name)
BankName	Name of bank to be shown on touch id / face id popup

2. Adding chatbot support to mobile application (Optional)

CHATBOT_ID	The tenant ID
CHATBOT_URL	The web socket URL for the ChatApp application in IBCS

3. Open constant.js file and in ssl status field enter the value "disabled" and in authenticator field enter the value "OBDXAuthenticator".



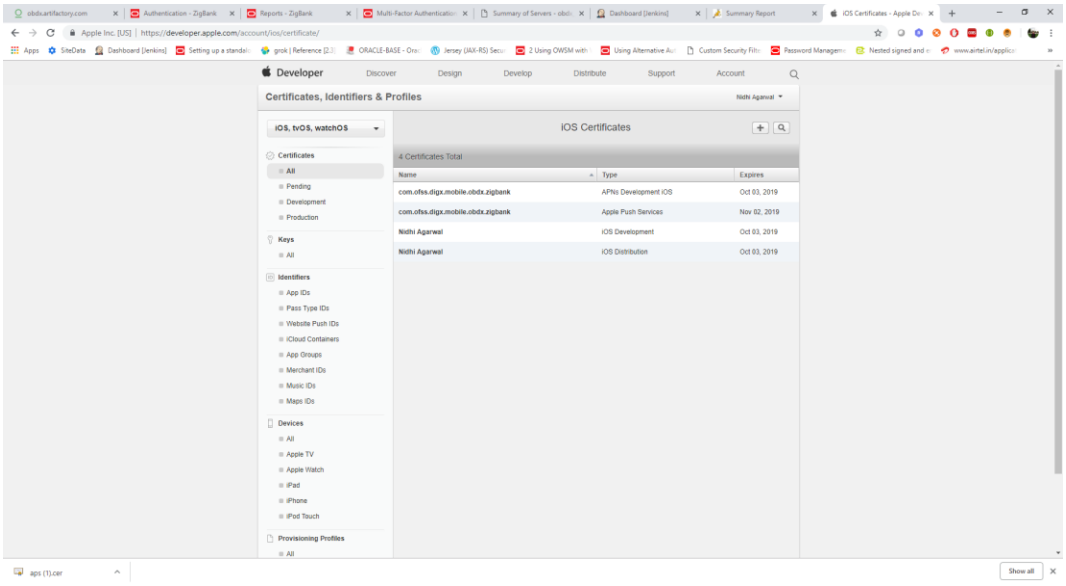
4. Adding Bundle Identifiers

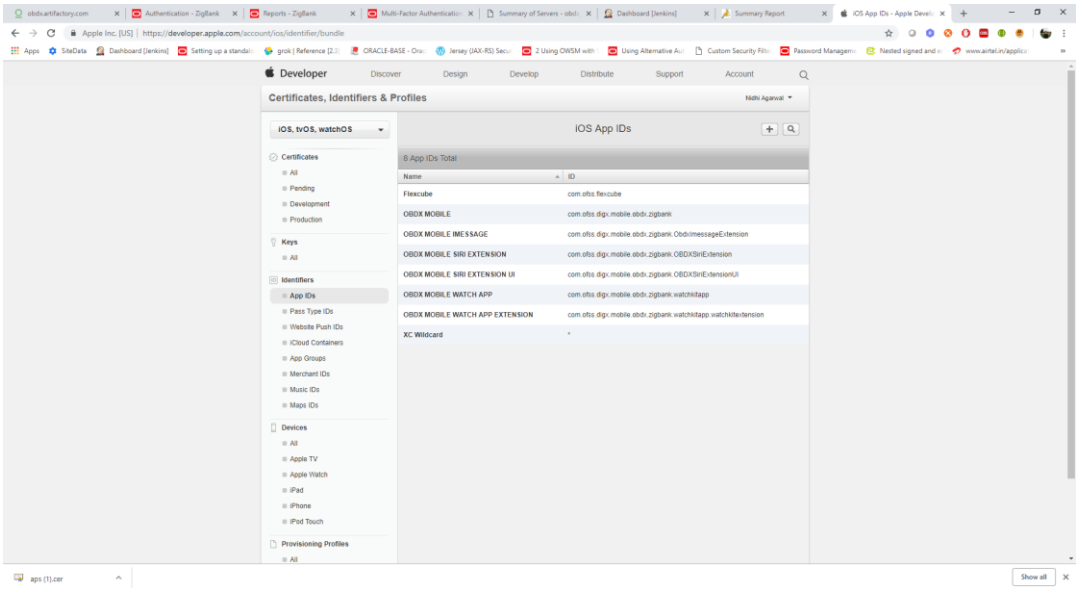
Bundle identifiers needs to be added in the Info.plist of each the frameworks along with the Signing Capabilities tab in Xcode. For example, the bundle identifier used is abc.def.ghi.jkl. The steps to be followed are,

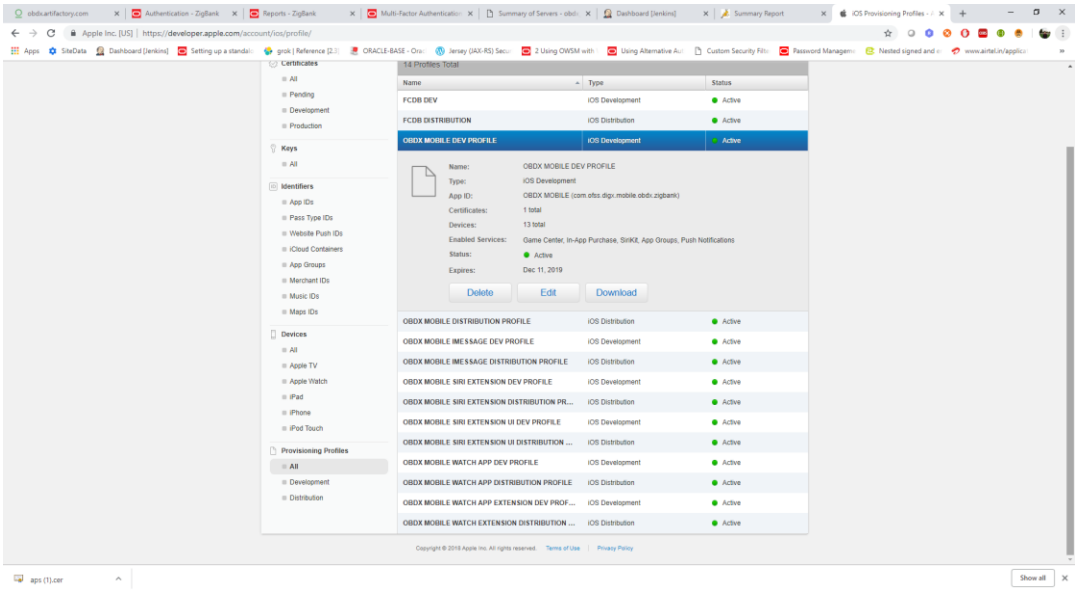
- Right click on OBDXFramework.framework(in Xcode's Project Navigator) -> Show in Finder
- When the finder directory opens the right click OBDXFramework.framework -> Show package contents.
- Open Info.plist and set Bundle identifier as abc.def.ghi.jkl.OBDXFramework
- Repeat the steps for the other three frameworks as well, with the following values:
    - Bundle identifier for Cordova.framework : abc.def.ghi.jkl.Cordova
    - Bundle identifier for OBDXExtensions.framework : abc.def.ghi.jkl.OBDXExtensions
    - Bundle identifier for OBDXWatchFramework.framework : abc.def.ghi.jkl. OBDXWatchFramework

2.5 Generating Certificates for Development, Production and Push Notifications

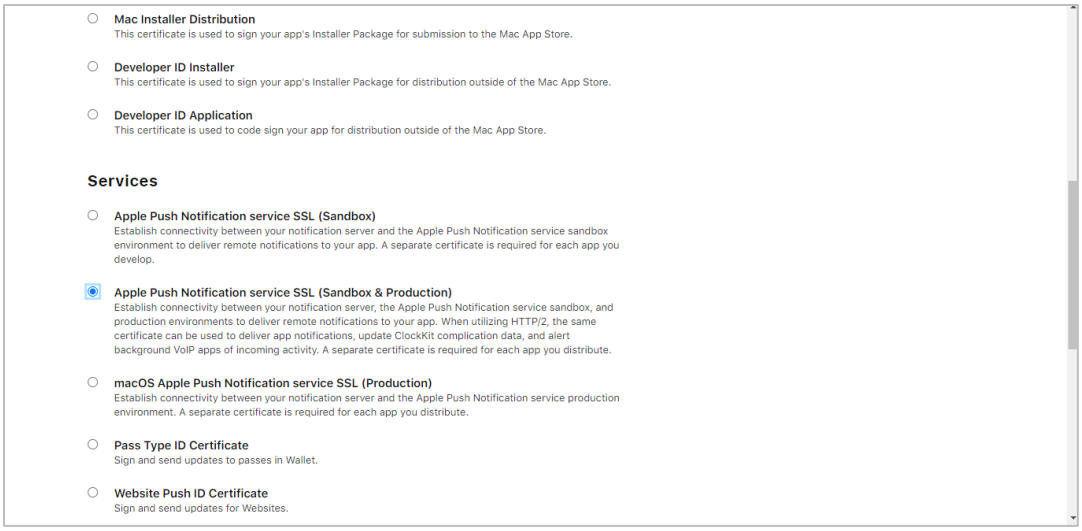
Create all certificates (by uploading CSR for keychain utility), provisioning profiles and push certificates as shown below by login in developer console. For development add device UUIDs and add same to provisioning profiles. Add capabilities as shown below and ensure the bundle identifier matches the one of the application in Xcode


Developer
Discover
Design
Develop
Distribute
Support
Account
Certificates, Identifiers & Profiles
iOS, tvOS, watchOS
iOS Certificates
4 Certificates Total
Name
Type
Expires
com.ofss.digx.mobile.obdx.zigbank
APNs Development iOS
Oct 03, 2019
com.ofss.digx.mobile.obdx.zigbank
Apple Push Services
Nov 02, 2019
Nidhi Agarwal
iOS Development
Oct 03, 2019
Nidhi Agarwal
iOS Distribution
Oct 03, 2019
Certificates
All
Pending
Development
Production
Keys
All
Identifiers
App IDs
Pass Type IDs
Website Push IDs
iCloud Containers
App Groups
Merchant IDs
Music IDs
Maps IDs
Devices
All
Apple TV
Apple Watch
iPad
iPhone
iPod Touch
Provisioning Profiles
All

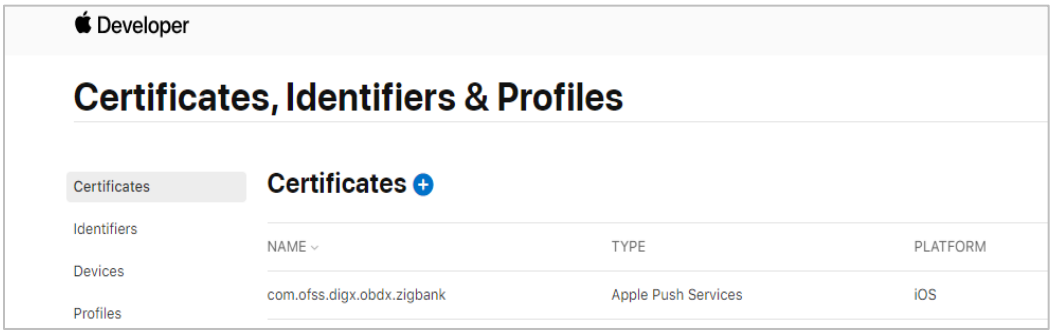

Developer
Discover
Design
Develop
Distribute
Support
Account
Certificates, Identifiers & Profiles
iOS, tvOS, watchOS
iOS App IDs
8 App IDs Total
Name
ID
Flexcube
com.ofss.flexcube
OBDX MOBILE
com.ofss.digx.mobile.obdx.zigbank
OBDX MOBILE IMESSAGE
com.ofss.digx.mobile.obdx.zigbank.ObdximessageExtension
OBDX MOBILE SIRI EXTENSION
com.ofss.digx.mobile.obdx.zigbank.OBDXSiriExtension
OBDX MOBILE SIRI EXTENSION UI
com.ofss.digx.mobile.obdx.zigbank.OBDXSiriExtensionUI
OBDX MOBILE WATCH APP
com.ofss.digx.mobile.obdx.zigbank.watchkitapp
OBDX MOBILE WATCH APP EXTENSION
com.ofss.digx.mobile.obdx.zigbank.watchkitapp.watchkitextension
XC Wildcard
*
Certificates
All
Pending
Development
Production
Keys
All
Identifiers
App IDs
Pass Type IDs
Website Push IDs
iCloud Containers
App Groups
Merchant IDs
Music IDs
Maps IDs
Devices
All
Apple TV
Apple Watch
iPad
iPhone
iPod Touch
Provisioning Profiles
All



Ensure AppGroups capability is added to all profiles and for mobile profile SiriKit, App Groups, Push Notifications must by added.



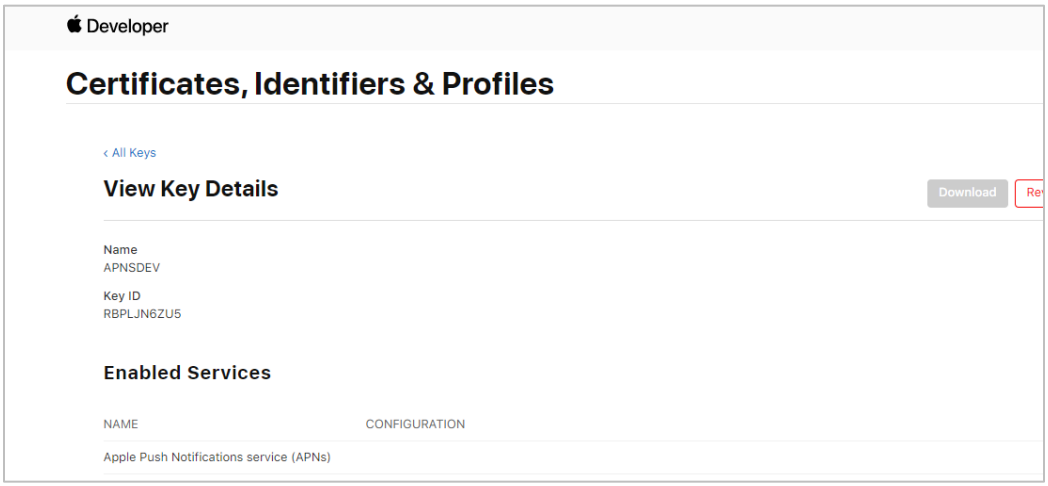
Note the certificate/bundle name



Note the Team ID from top right corner

Navigate to the “Keys” section and create APNS key

Note APNS key and download the .p8 file. Copy the .p8 to config/resources\mobile

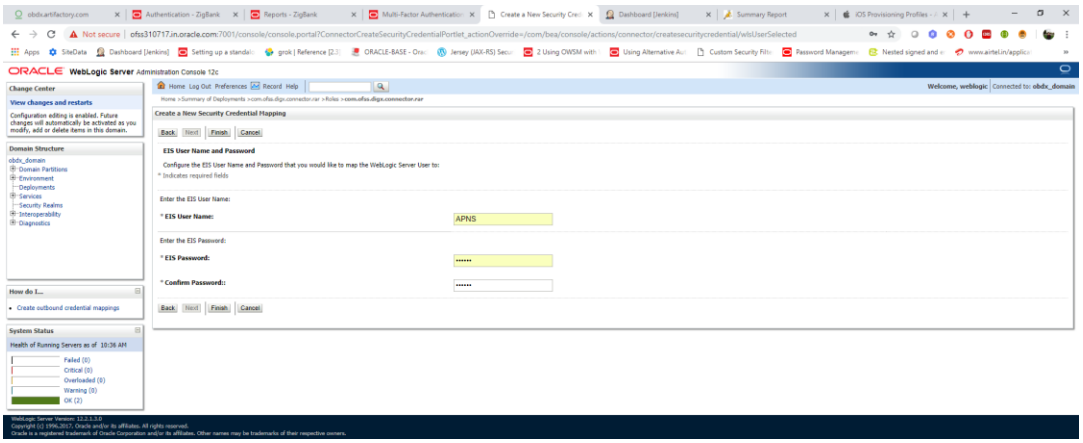


Update the password as shown below –

Sr. No.	Table	PROP_ID	CATEGORY_ID	PROP_VALUE	Purpose
1	DIGX_FW_CONFIG_ALL_B	ios_cert_path	DispatchDetails	resources/mobile/AuthKey_RBPLJN6ZU5.p8	Update the certificate path/name if required. Should be relative to config directory
2	DIGX_FW_CONFIG_ALL_B	APNS	DispatchDetails	<Password> Eg - RBPLJN6ZU5	Provides id of .p8 certificate
3	DIGX_FW_CONFIG_ALL_B	APNSKeyStore	DispatchDetails	DATABASE or CONNECTOR	Specifies whether to pick certificate password from database or from connector. Default DB (No change)
4	DIGX_FW_CONFIG_ALL_B	Proxy	DispatchDetails	<protocol,proxy_address>	Provides proxy address, if any, to be provided while connecting to APNS server. Delete row if proxy not required. Example: HTTP,148.50.60.1,80

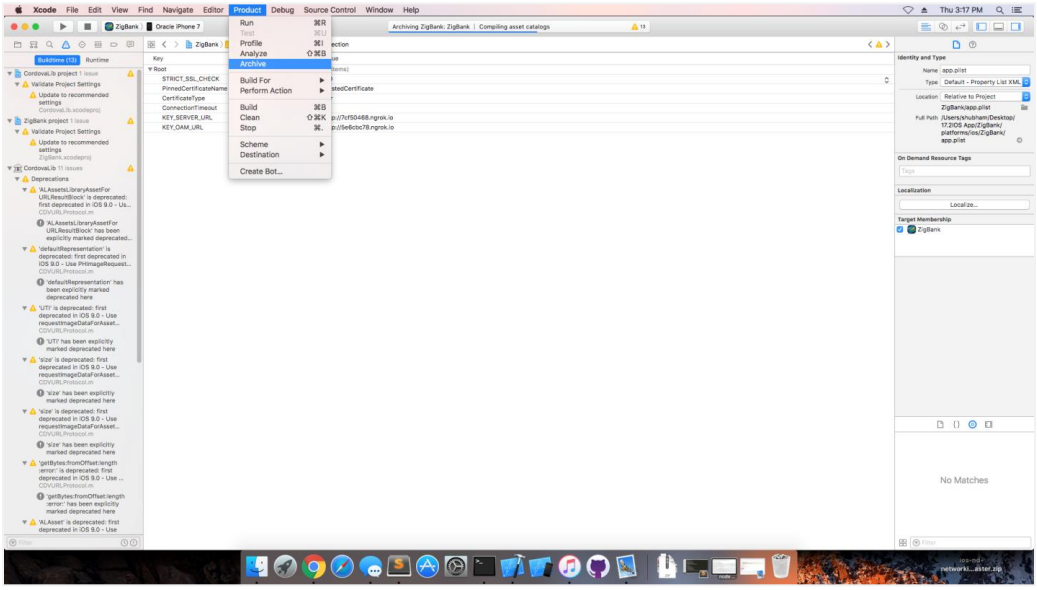
5	DIGX_FW_CONFIG_ALL_B	CERT_TYPE	DispatchDetails	For dev push certs add row with value ‘dev’	For prod push certificates this row is not required
6	DIGX_FW_CONFIG_ALL_B	APNS_BUNDLE	DispatchDetails	Eg. com.ofss.digx.obdx.zigbank	Bundle Name
7	DIGX_FW_CONFIG_ALL_B	APNS_TEAMID	DispatchDetails	Eg. 3NX1974C93	Team ID of Apple developer account

If CONNECTOR is selected in Step 2 update certificate id as below

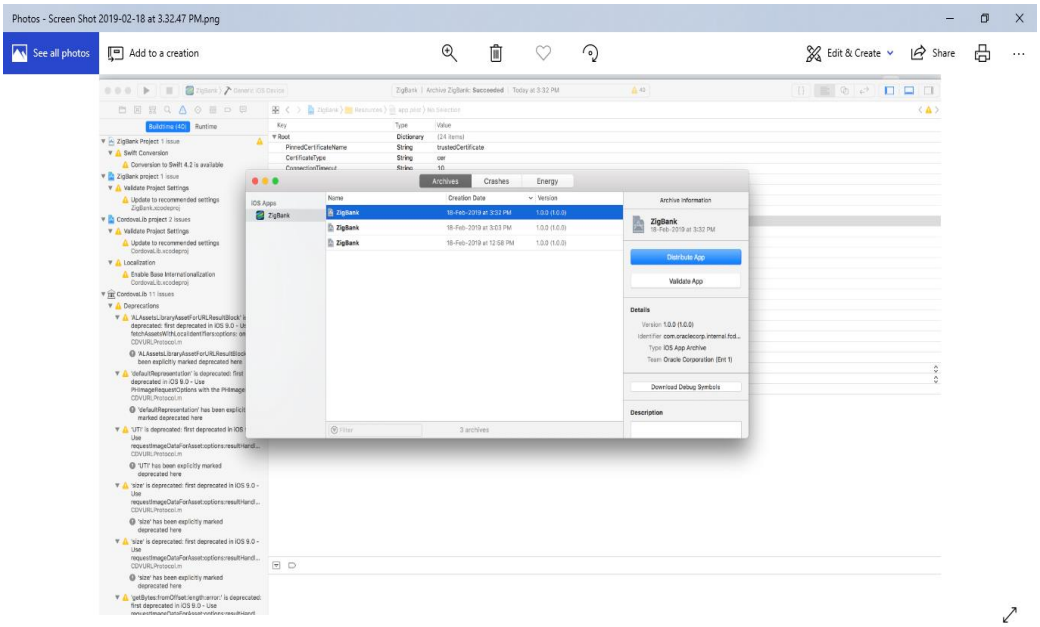


3. Archive and Export

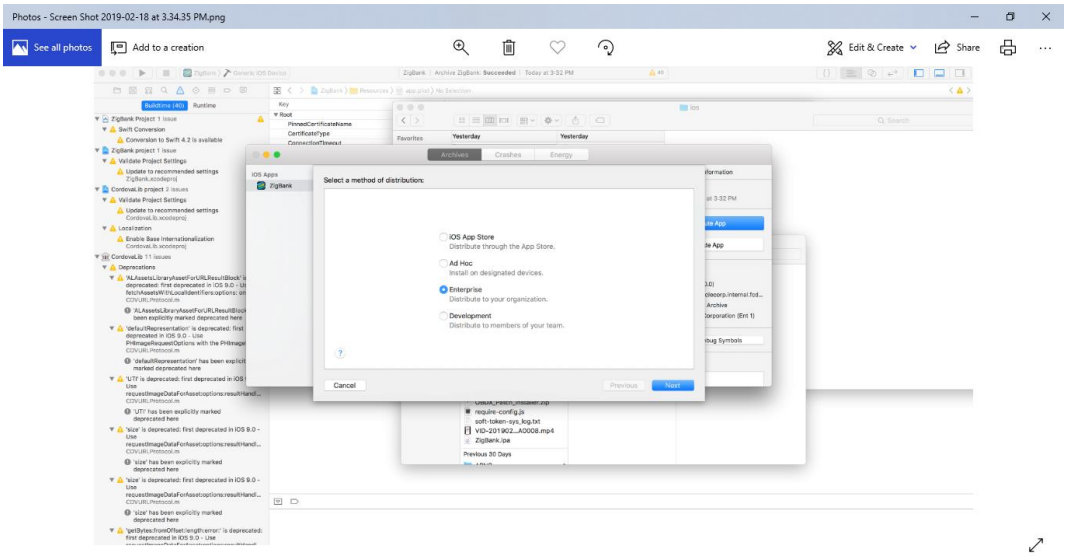
a. In the Menu bar click on **Product -> Archive (Select Generic iOS Device)**

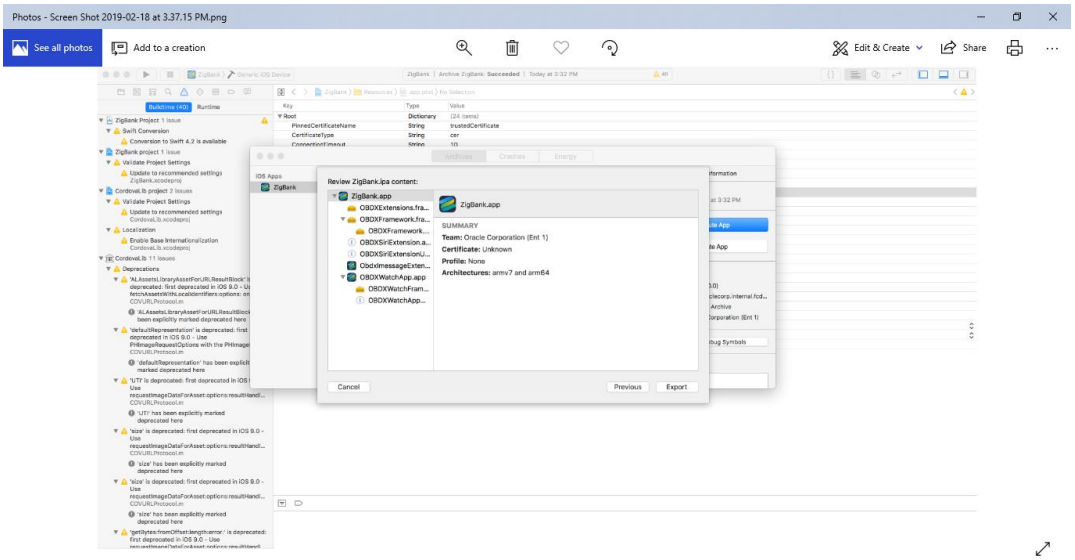


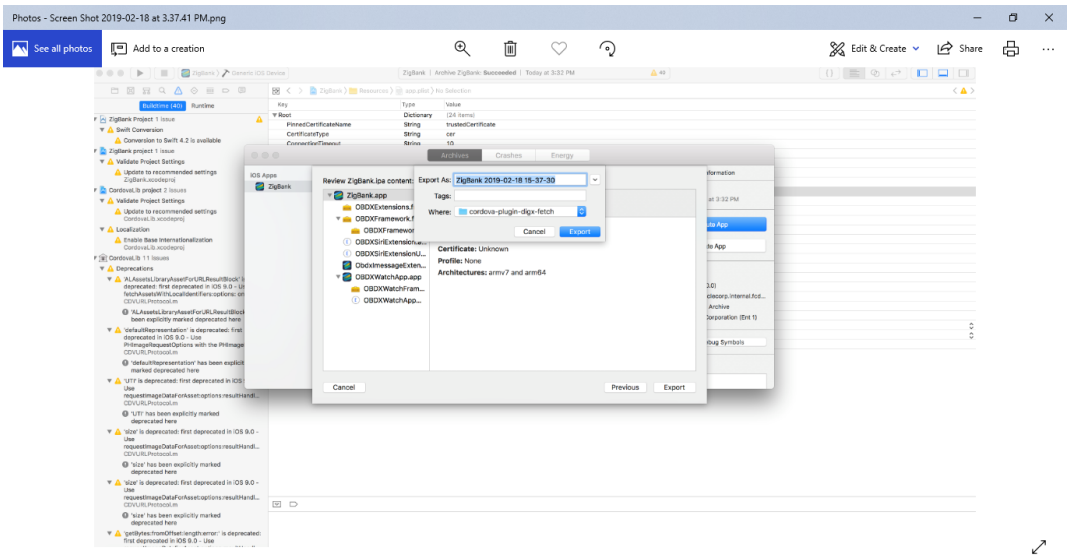
b. After archiving has successfully completed. Following popup will appear



c. Click on Distribute App in the right pane of the popup -> select **the Method of Distribution -> Choose Provisioning Profile** according to the method of distribution **->** select **Next ->** Review the contents and click on **Export -> Export** and generate the .ipa


Photos - Screen Shot 2019-02-18 at 3.34.35 PM.png
See all photos
Add to a creation
Edit & Create
Share
Select a method of distribution:
iOS App Store
Distribute through the App Store.
Ad Hoc
Install on designated devices.
Enterprise
Distribute to your organization.
Development
Distribute to members of your team.
Cancel
Previous
Next


Photos - Screen Shot 2019-02-18 at 3.37.15 PM.png
See all photos
Add to a creation
Edit & Create
Share
Review ZigBank.ipa content:
ZigBank.app
SUMMARY
Team: Oracle Corporation (Ent 1)
Certificate: Unknown
Profile: None
Architectures: armv7 and arm64
Cancel
Previous
Export



To run the application on simulator copy & replace 3 frameworks (.framework files) from /simulator to zigbank/platforms/ios/

4. OBDX Authenticator Application

4.1 Authenticator UI (Follow any one step below)

4.1.1 Using built UI

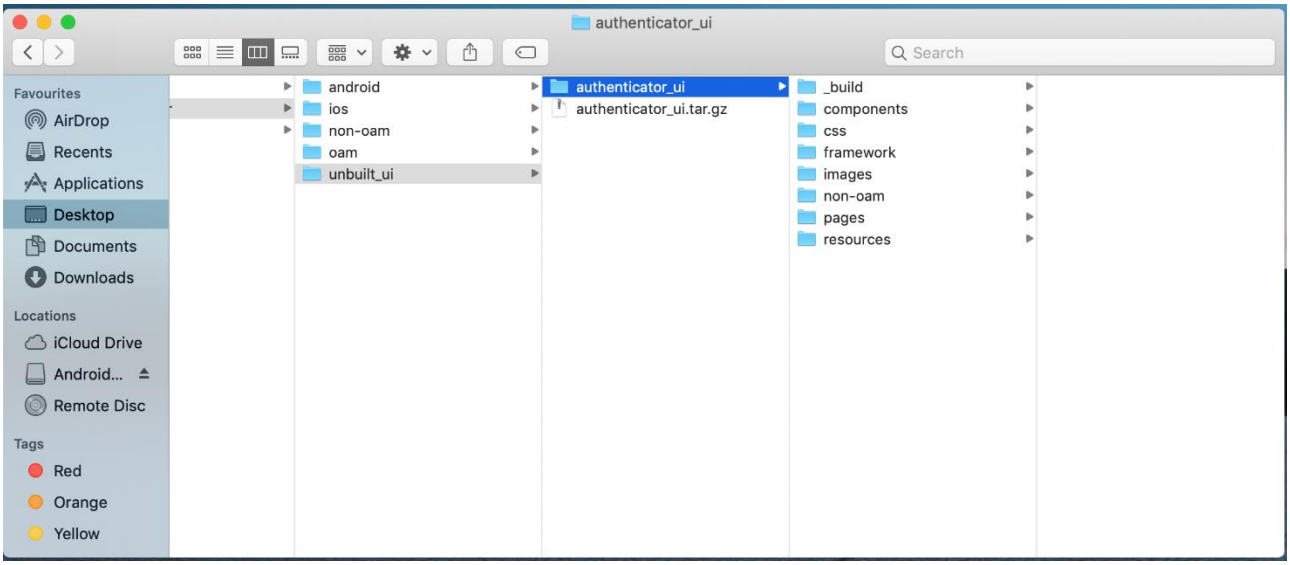
For Non-OAM - Unzip dist.tar.gz directory from OBDX_Patch_Mobile\authenticator\non-oam

For OAM - Unzip dist.tar.gz directory from OBDX_Patch_Mobile\authenticator\oam

4.1.2 Building UI manually

1. Extract authenticator_ui.tar.gz from OBDX_Patch_Mobile\authenticator\unbuilt_ui.

   The folder structure is as shown :



(a) OAM based Authentication

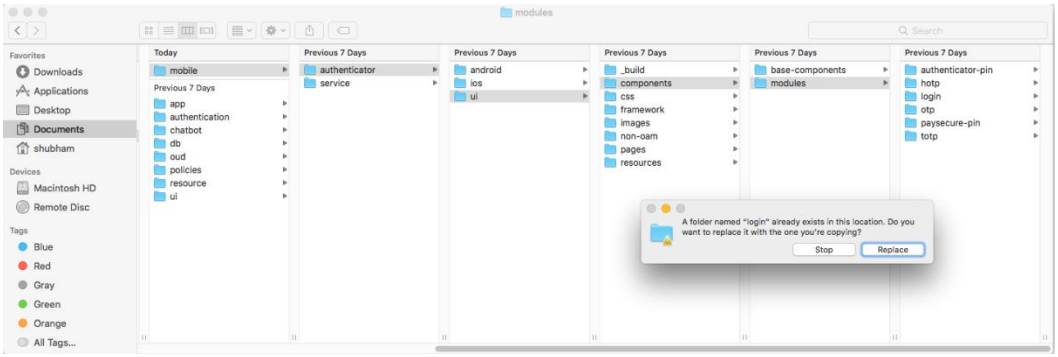
1. Open Terminal at *"_build"* level.
2. Run following command :

```
sudo npm install -g grunt-cli

sudo npm install

node render-requirejs/render-requirejs.js

grunt authenticator --verbose
```

3. After running above commands and getting result as "*Done, without errors.*" a new folder will be created at "_build" level with name as "dist".

(b) NON-OAM Based Authentication

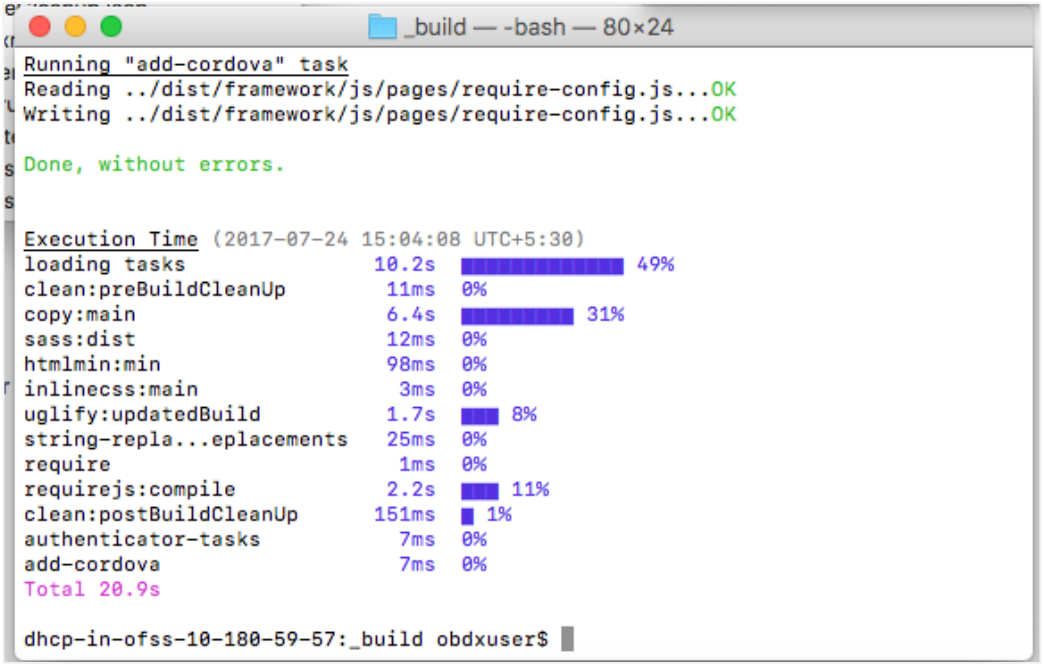
1. Copy "*non-oam/login"* folder and Replace it at location "*components/modules/" [in ui folder]* location. This will replace existing "*login*" folder.

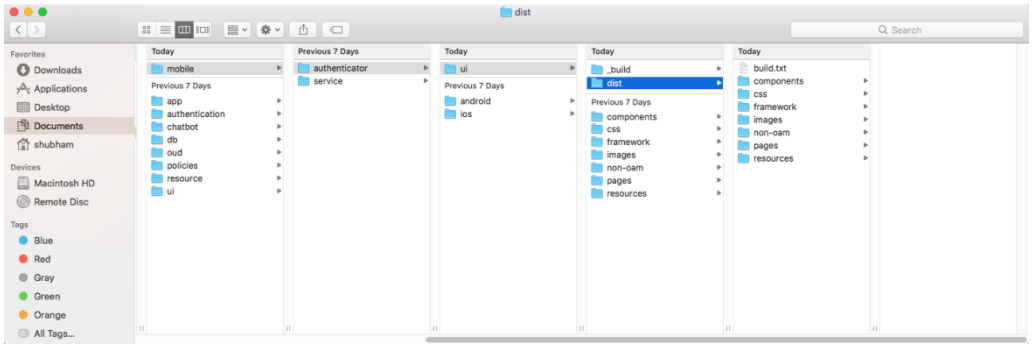


2. Open Terminal at "_build" level.
3. Run following command :

```
sudo npm install -g grunt-cli

sudo npm install

node render-requirejs/render-requirejs.js

grunt authenticator --verbose
```

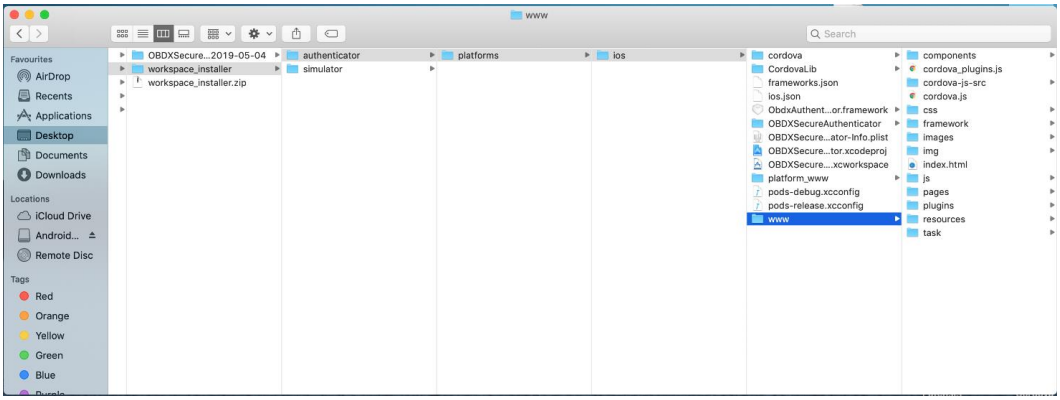
4. After running above commands and getting result as "*Done, without errors.*" a new folder will be created at "_build" folder level with name as "dist".


_build — -bash — 80×24
Running "add-cordova" task
Reading ../dist/framework/js/pages/require-config.js...OK
Writing ../dist/framework/js/pages/require-config.js...OK
Done, without errors.
Execution Time (2017-07-24 15:04:08 UTC+5:30)
loading tasks 10.2s 49%
clean:preBuildCleanUp 11ms 0%
copy:main 6.4s 31%
sass:dist 12ms 0%
htmlmin:min 98ms 0%
inlinecss:main 3ms 0%
uglify:updatedBuild 1.7s 8%
string-repla...eplacements 25ms 0%
require 1ms 0%
requirejs:compile 2.2s 11%
clean:postBuildCleanUp 151ms 1%
authenticator-tasks 7ms 0%
add-cordova 7ms 0%
Total 20.9s
dhcp-in-ofss-10-180-59-57:_build obdxuser$

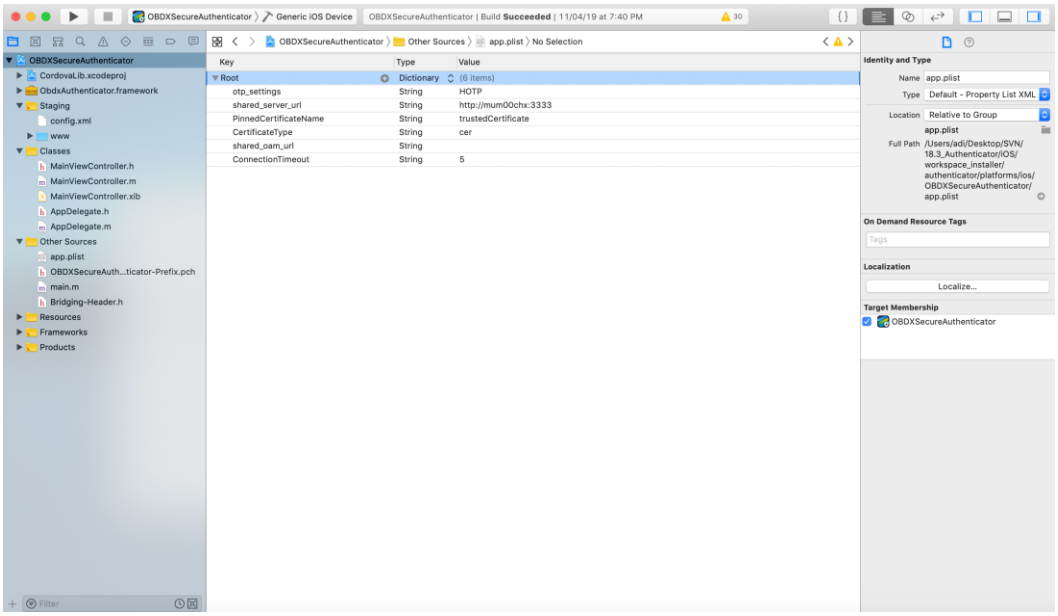

dist
Favorites
Downloads
Applications
Desktop
Documents
shubham
Devices
Macintosh HD
Remote Disc
Tags
Blue
Red
Gray
Green
Orange
All Tags...
Today
mobile
Previous 7 Days
app
authentication
chatbot
db
oud
policies
resource
ui
authenticator
service
android
ios
_build
dist
components
css
framework
images
non-oam
pages
resources
build.txt

4.2 Authenticator Application Workspace Setup

1. Unzip and navigate to iOS workspace as shipped in installer.
2. Open the workspace as shown below and find and replace the following generated UI files from "*ui/dist*" folder :
   - components
   - css
   - framework
   - images
   - pages
   - resources



3. Double click on OBDXSecureAuthenticator.xccodeproj to open the project in Xcode



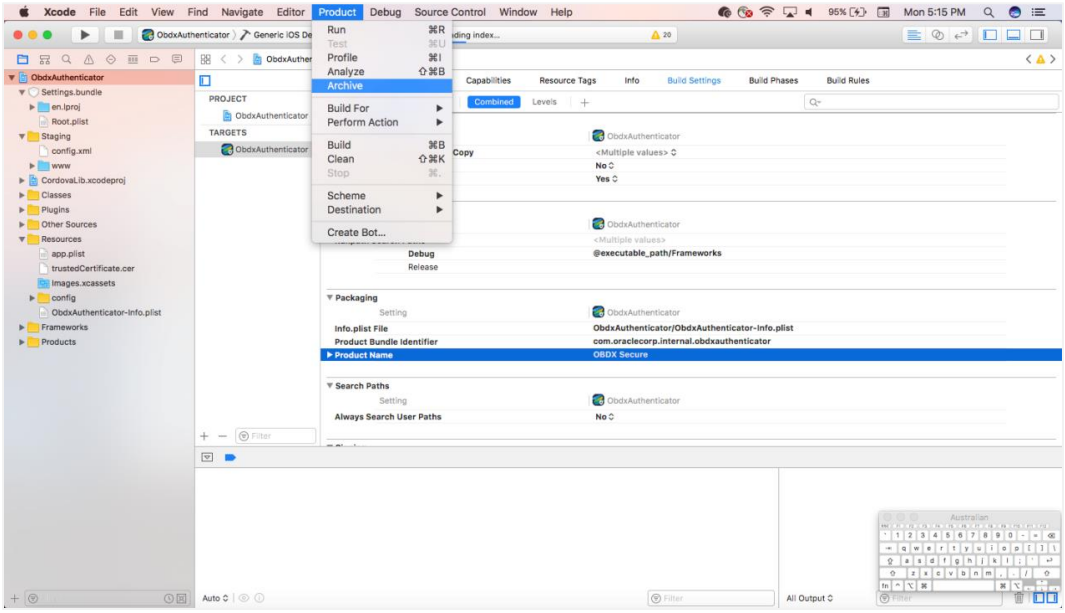
Update HOTP or TOTP in above screenshots and update the server URL.

4. The application can be archived using steps in Section 4.3 for running on device

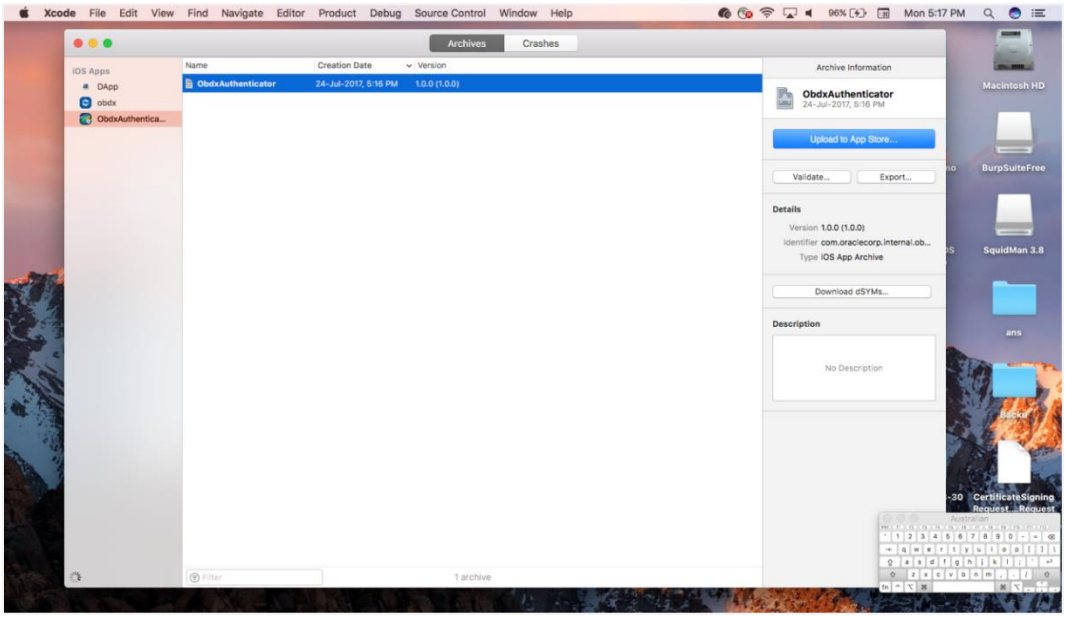
5. To run the application on simulator, copy & replace the framework from simulator/ObdxAuthenticator.framework to /authenticator/platforms/ios/

4.3 Building Authenticator Application

1. Set the simulator to *Generic iOS* device. Then go to *Product -> Archive*.



2. Choose your Archive and then click "*Export*". .ipa file will be generated



5. Application Security Configuration

OBDX supports single and multiple certificate pinning as part of application security. This can be achieved by maintaining below key values in app.plist for both the apps

1. PinnedCertificateName - Certificate name (can be multiple in case of OAM)
2. PinnedUrl – URLs to be pinned
3. CertificateType – cer/der